

**IN THE CLAIMS**

For the convenience of the Examiner, all pending claims of the present Application are presented below whether or not an amendment has been made. Please amend the claims as follows:

1. **(Currently Amended)** A method of arranging data in a database comprising:

creating a first table adapted for storing data comprising at least one data entry, the data entry comprising a plurality of data components, the first table comprising one row for each data entry; and

~~using the first table to create~~ creating a second table ~~adapted for~~ storing the plurality of data components of the data entry of the first table, the second table comprising one row for each of the plurality of data components of the data entry of the first table.

2. **(Original)** The method as claimed in claim 1, wherein the data is a structured data type.

3. **(Original)** The method as claimed in claim 1, wherein the data is a string data type.

4. **(Original)** The method as claimed in claim 1, wherein the data is or represents a X.509 certificate.

5. **(Previously Presented)** The method as claimed in claim 1, wherein a selected one of the data components is a checksum or fingerprint.

6. **(Original)** The method as claimed in claim 1, where the database is a part of an electronic directory services system.

7. **(Original)** The method as claimed in claim 6, where the electronic directory services system comprises an X.500 and LDAP services system.

8. **(Currently Amended)** A database having a data storage arrangement comprising:

a search table comprising at least one row having a plurality of columns, each column of the at least one row storing a data component; and

a subsearch table ~~created from the search table, the subsearch table~~ comprising one row for each data component of the at least one row of the search table, each row having a plurality of columns including a component identifier column configured to be used as a search index for searching data components in the at least one row of the search table.

9. **(Original)** The database as claimed in claim 8, wherein the columns of the search table are in the form "EID, AID, VID, Norm", where EID identifies an object to which a value belongs, AID identifies an attribute type of the value, and VID identifies one of a possible number of attribute values in the one entry.

10. **(Original)** The database as claimed in claim 8, wherein the columns of the search table are in the form "EID, AID, VID, CID, Norm", where EID identifies an object to which a value belongs, AID identifies an attribute type of the value, VID identifies one of a possible number of attribute values in the one entry, and CID identifies the component identifier.

11. **(Original)** The database as claimed in claim 8, further comprising a subattribute table containing at least one row having a plurality of columns in which a description or reference to the subsearch table is provided.

12. **(Original)** The database as claimed in claim 11, wherein the columns of the subattribute table are in the form "CID, SYN, DESC, OBJECT ID, FLAGS".

13. **(Currently Amended)** A database having a data storage arrangement comprising:

a first table directed to a hierarchy which defines a relationship between a plurality of objects and configured to have one row per object;

a second table ~~created from the first table and~~ directed to the plurality of objects of the first table, the second table defining one or more values within each of the plurality of objects of the first table and configured to have one row per value; and

a third table directed to one or more selected components of the one or more values of the second table and configured to have one row for each component of each of the one or more values of the second table.

14. **(Currently Amended)** A directory services system for performing directory service requests on a database, comprising:

a first table arranged for storing data comprising at least one data entry, the data entry comprising a plurality of data components, the first table comprising one row for each data entry; and

a second table ~~created from the first table, the second table arranged for~~ storing the plurality of data components of the data entry of the first table, the second table comprising one row for each of the plurality of data components of the data entry of the first table.

15. **(Original)** A directory services system as claimed in claim 14, wherein the data is a structured data type.

16. **(Original)** A directory services system as claimed in claim 14, wherein the data is a string data type.

17. **(Original)** The directory services system of claim 14, being an X.500 or LDAP directory services system.

18. **(Currently Amended)** A directory services system having a data storage arrangement comprising:

a first table directed to a hierarchy which defines a relationship between a plurality of objects and configured to have one row per object;

a second table ~~created from the first table and~~ directed to the plurality of objects of the first table, the second table defining one or more values within each of the plurality of objects of the first table and configured to have one row per value; and

a third table directed to one or more selected components of the one or more values of the second table and configured to have one row for each component of each of the one or more values of the second table.

19. **(Original)** A directory services system as claimed in claim 18, wherein the data is a structured data type.

20. **(Original)** A directory services system as claimed in claim 18, wherein the data is a string data type.

21. **(Original)** The directory services system of claim 18, being an X.500 or LDAP directory services system.

22. **(Previously Presented)** A method of searching a database for given data entries, comprising:

determining a component of a given data entry of a first table, the given data entry of the first table comprising a plurality of data components;

identifying a component identifier indicating a data type that is associated with the component of the first table;

using the component identifier indicating the data type to execute one of an exact or initial matching on a column of a second table in order to locate the component in the second table, the second table comprising one row for each of the plurality of data components of the given data entry of the first table; and

returning the given data entry from the first table matching the component located.

23. **(Original)** The method as claimed in claim 22, where the database is a part of an electronic directory services system.

24. **(Original)** The method as claimed in claim 22, where the electronic directory services system comprises an X.500 and LDAP services system.

25. **(Original)** The method as claimed in claim 22, wherein the data is or represents a X.509 certificate, and / or a check sum of the data and / or a fingerprint of the data.

26. **(Original)** The method as claimed in claim 23, wherein the component is a checksum or fingerprint of the data.

27. **(Original)** The method as claimed in claim 26, wherein the search is conducted using a search table to locate the fingerprint or checksum.

28. **(Original)** A method as claimed in claim 27, further wherein components of the checksum or fingerprint are searched.